

Simulation Results of User Behavior-Aware Scheduling Based on Time-Frequency Resource Conversion

Hanguan Shan¹, Yani Zhang¹, Weihua Zhuang², Aiping Huang¹, and Zhaoyang Zhang¹

¹College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China

²Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Canada

Abstract

Integrating time-frequency resource conversion (TFRC), a new network resource allocation strategy, with call admission control can not only increase the cell capacity but also reduce network congestion effectively. However, the optimal setting of TFRC-oriented call admission control suffers from the curse of dimensionality, due to Markov chain-based optimization in a high-dimensional space. To address the scalability issue of TFRC, in [1] we extend the study of TFRC into the area of scheduling. Specifically, we study downlink scheduling based on TFRC for an LTE-type cellular network, to maximize service delivery. The service scheduling of interest is formulated as a joint request, channel and slot allocation problem which is NP-hard. An offline deflation and sequential fixing based algorithm (named DSFRB) with only polynomial-time complexity is proposed to solve the problem. For practical online implementation, two TFRC-enabled low-complexity algorithms, modified Smith ratio algorithm (named MSR) and modified exponential capacity algorithm (named MEC), are proposed as well. In this report, we present detailed numerical results of the proposed offline and online algorithms, which not only show the effectiveness of the proposed algorithms but also corroborate the advantages of the proposed TFRC-based schedule techniques in terms of quality-of-service (QoS) provisioning for each user and revenue improvement for a service operator.

Keywords: Time-frequency resource conversion, context-aware resource allocation, scheduling.

I. SCHEDULING PERFORMANCE

To evaluate the proposed TFRC-enabled scheduling techniques, extensive simulations are performed for both offline and online scheduling. For offline scheduling, we focus on the performance improvement of TFRC-enabled schedule and the benefits of the new penalty function proposed in [1], while for online scheduling we measure the performance gap between the proposed offline and multiple online algorithms.

Specifically, for offline scheduling, two benchmark algorithms are compared with the proposed DSFRB algorithm: the algorithm proposed in [2] and integrated with the new penalty function (denoted as “DSF-NP”) and the algorithm with the traditional penalty function (denoted as “SF-OP”)¹. For online scheduling, two other benchmark algorithms are compared with the proposed MSR and MEC algorithms: the algorithm with earliest-deadline-first policy (denoted as “EDF”) and the algorithm proposed in [3] (denoted as “L-MaxWeight”). L-MaxWeight is tailored for scheduling flows with deadlines and is shown superior performance in underloaded identical-deadline systems. For either scenario, the impacts of different parameters, including traffic load, request size, mean request lifetime, and channel condition, on scheduling performance are studied.

In the simulation, we assume that there are 5 users being served by a cell with 32 subchannels, each channel with a bandwidth of 15 KHz. The channel fading is modeled by the Rayleigh distribution. Each user initiates new requests according to a Poisson process with rate λ_u . The data size Q_k and lifetime Π_k of the k^{th} request are uniformly distributed within $[Q_{\min}, Q_{\max}]$ and exponentially distributed with mean μ , respectively. The reward of the request’s unit data (A_k) is assumed to be uniformly distributed within $[1, 10]$. The context information on user behavior is fed back from user equipment (UE) to the base station (BS) once every slot of 100 ms. For all simulation results, we perform the simulation for 200 runs, average the results, and obtain the 95% confidence intervals.

A. Offline Scheduling

Fig. 1 shows the impact of traffic load on both total system reward (operator side) and complete ratio of all requests (user side), with $\lambda_u \in [0.01, 0.1]$ requests/s, $Q_{\min} = 15$ Mbits, $Q_{\max} = 20$ Mbits, $\mu = 30$ s, and mean signal-to-noise ratio (SNR) for each channel equal to 10 dB. Unless otherwise specified, simulation for effects of other parameters are all based on the same setting, and the changed parameters are listed. In Fig. 1, both results for the TFRC-enabled and TFRC-disabled DSFRB, DSF-NP, and SF-OP algorithms are presented. It is observed that the algorithms enabling TFRC outperform their counterpart disabling TFRC, in terms of the system reward and

¹Scheduling with the traditional penalty function is to solve a mixed integer linear OP (see Proposition 2 of [1]). So, we can solve the OP directly with the SF algorithm proposed in [1] effectively.

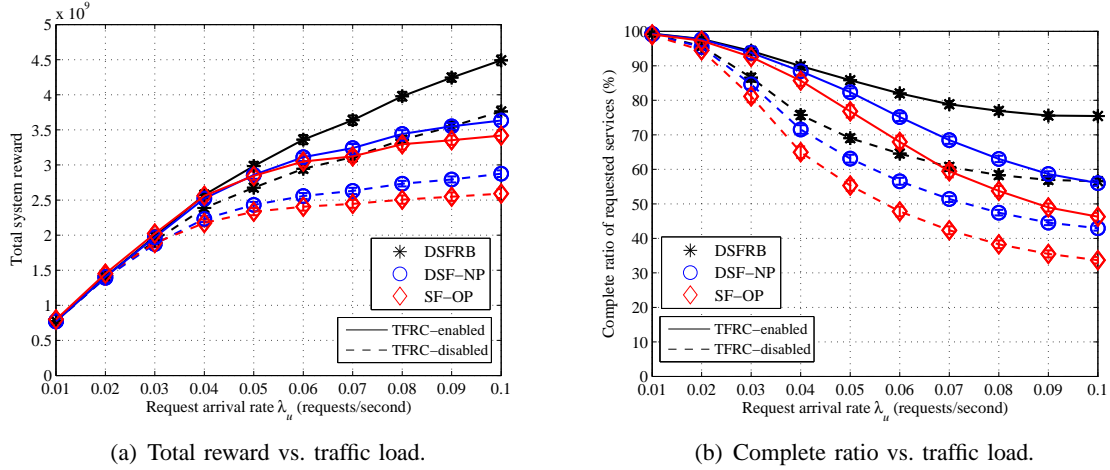


Fig. 1. Impact of traffic load on total system reward and complete ratio with offline scheduling.

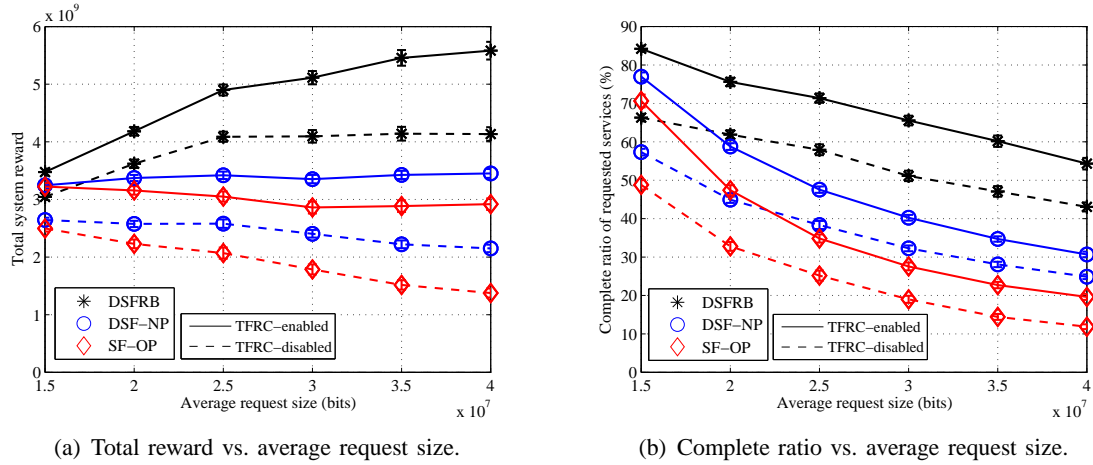


Fig. 2. Impact of average request size on total system reward and complete ratio with offline scheduling.

complete ratio, generating a potential win-win situation for both the operator and the user. The performance improvement results from the fact that the context information (which connection(s) is currently focused on by the user) utilized in the TFRC-enabled schedule offers more freedom to find a better scheduling solution. Further, the performance gap, especially the one for DSFRB, increases with traffic load, illustrating the potential advantage of TFRC-enabled schedule in addressing a heavy-load network scenario. In Fig. 1, a performance gap between DSFRB and DSF-NP exists, mainly due to the newly proposed deflation strategy. Both DSFRB and DSF-NP outperform SF-OP, illustrating an advantage of the new penalty function over the traditional one in increasing delivered request number.

Fig. 2 shows how the total system reward and the complete ratio change with the average request size, with $\lambda_u = 0.075$ requests/s, $Q_{\min} = 10$ Mbits, and Q_{\max} changing with the average

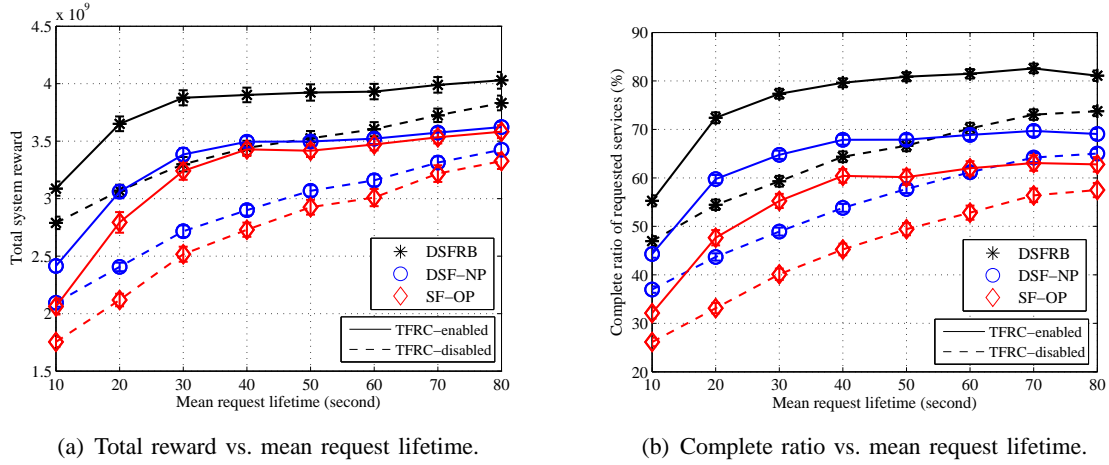


Fig. 3. Impact of mean request lifetime on total system reward and complete ratio with offline scheduling.

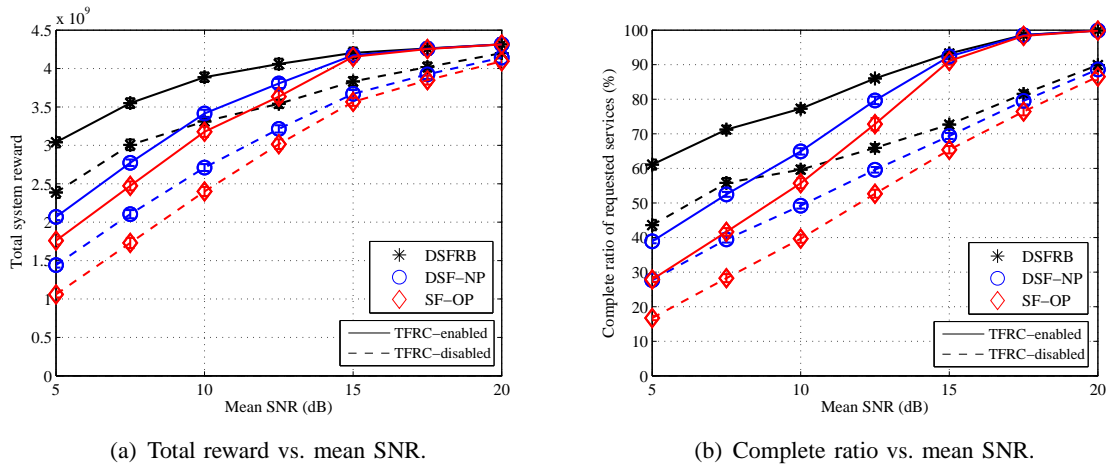


Fig. 4. Impact of mean SNR on total system reward and complete ratio with offline scheduling.

request size from 15 Mbits to 40 Mbits. We can see that the complete ratio of request services with any of the three algorithms decreases with an increase of the average request size, as the traffic load increases with the average request size. The TFRC strategy makes each algorithm avoid suffering too much from the increased traffic load, as observed in Fig. 1. Further, it is observed in Fig. 2(a) that, without applying TFRC, the total system reward for DSF-NP or SF-OP decreases with the average request size; however, the performance of algorithms enabling TFRC remains almost unchanged. On the other hand, the total system reward for DSFRB without TFRC increases slightly for the average request sizes; yet, a much larger increasing rate is observed for the algorithm enabling TFRC.

In Fig. 3, we study the impact of mean request lifetime on the total system reward and the complete ratio of all requests, with $\lambda_u = 0.075$ requests/s and $\mu \in [10, 80]$ s. It is observed

that both total system reward and complete ratio of the three algorithms, with or without TFRC, increase as the mean lifetime μ increases. With an increase of μ , each request has more time for resource allocation, thus more chances to be delivered before deadline. Yet, for both performance metrics, DSFRB is much better than other two algorithms, due to the same reasons as aforesaid. Also, by applying TFRC, all algorithms perform better but tend to converge when $\mu \geq 40$ s. This is because the cell capacity has been fully exploited by the TFRC strategy when $\mu \geq 40$ s.

Fig. 4 shows how the total system reward and the complete ratio of requested services change with the channel quality. Here, we set $\lambda_u = 0.075$ requests/s, vary the mean SNR of each channel from 5 dB to 20 dB, and keep other parameters the same as for Fig. 1. Both performance metrics improve with the channel quality because the data transmission rate increases with it as well. However, it is clear that integrating TFRC with scheduling techniques helps each algorithm harvest much more potential benefits from the improved channel quality.

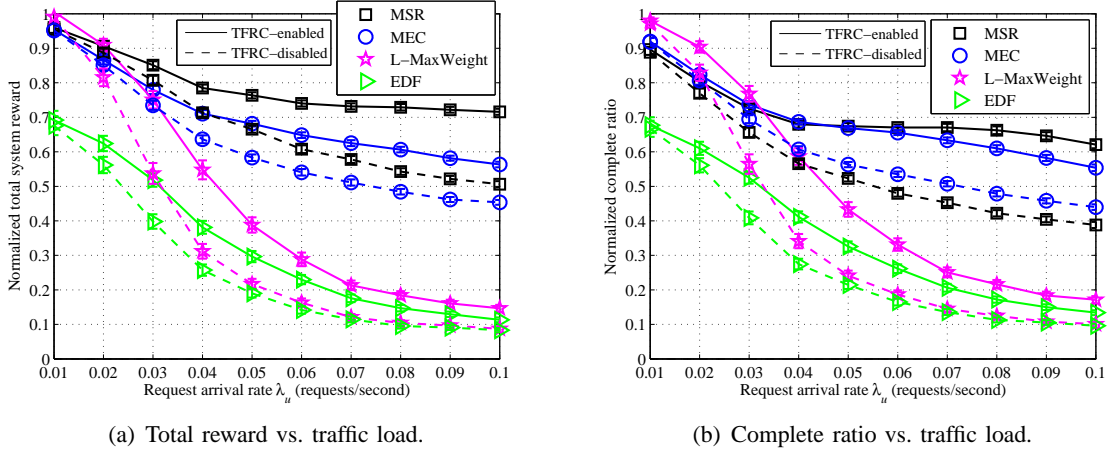


Fig. 5. Impact of traffic load on total system reward and complete ratio with online scheduling.

B. Online Scheduling

The good performance of TFRC-enabled schedule benefits from not only algorithm design but also non-causal information on request demand and channel capacity (see [1]). Next, we evaluate its online counterpart, thus to understand the effect of TFRC-enabled scheduling techniques in a more practical scenario. In the following, all simulation settings are the same as those for offline scheduling. The results are normalized by TFRC-enabled DSFRB algorithm.

Fig. 5 shows the impact of traffic load. It is observed that each algorithm performs better if enabling TFRC. As the traffic load increases, the normalized total system reward with any of the

compared algorithms decreases, and the performance gap between these online algorithms and the DSFRB algorithm increases. However, from the simulations we find that the total system reward (i.e., the absolute value) with MSR or MEC always increases with the traffic load. Further, the two algorithms perform much better than both L-MaxWeight and EDF, especially when traffic load is heavy. For example, when enabling TFRC and $\lambda_\mu = 0.1$ requests/s, as compared with L-MaxWeight the improvement of total system reward (complete ratio) of MSR and MEC can be 387.9% and 283.8% (262.1% and 223.0%), respectively. However, TFRC-enabled L-MaxWeight performs slightly better than MSR and MEC when request arrival rate is less than 0.02 requests/s, showing the advantage of L-MaxWeight in a low traffic load region.

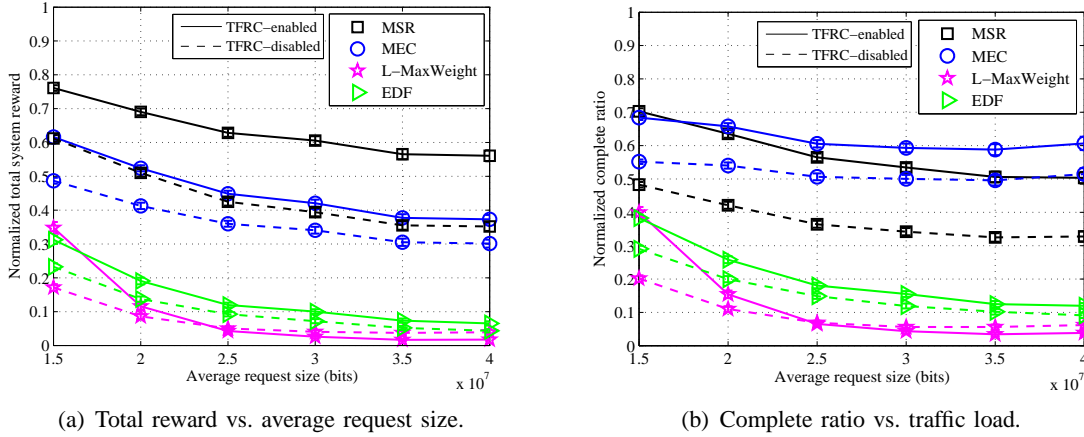


Fig. 6. Impact of average request size on total system reward and complete ratio with online scheduling.

Fig. 6 evaluates the effect of request size. The MSR and MEC outperform EDF and L-MaxWeight in both total system reward and complete ratio, as the average request size increases. However, the normalized total system reward or complete ratio for each of the four algorithms reduces as the average request size increases, because of not only the increased traffic load but also the different scheduling capabilities of these algorithms as compared with the offline DSFRB. Integrating with TFRC helps MSR, MEC, and EDF achieve much better performance than their TFRC-disabled counterparts, which does not hold for L-MaxWeight when average request size is larger than 2.5 Mbits. The reason is that, with user behavior information, TFRC allows a request to be scheduled in a longer time, which also has an effect of accumulating network traffic load. The potentially increased traffic load generates a negative impact on L-MaxWeight.

Fig. 7 studies the impact of mean request lifetime. It can be seen that the performance gap between the online algorithms and the offline DSFRB algorithm reduces with mean request

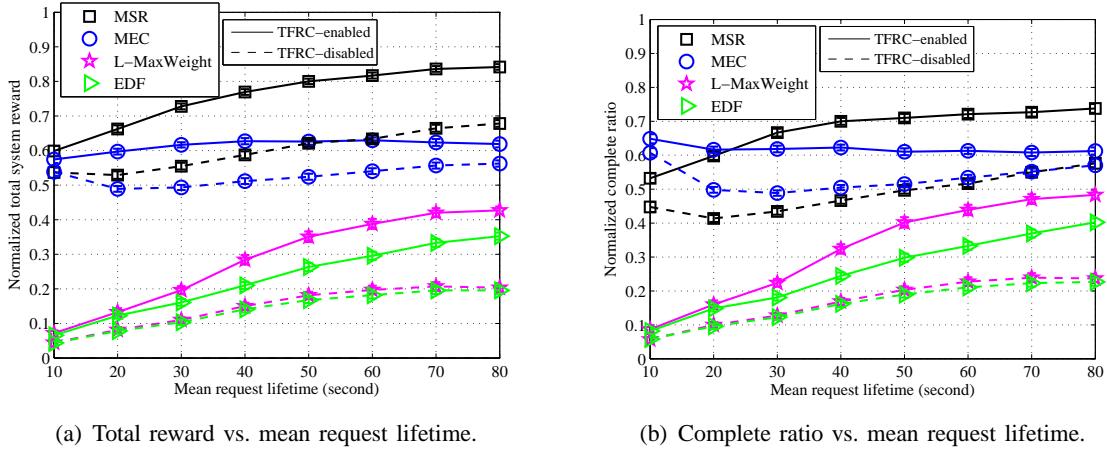


Fig. 7. Impact of mean request lifetime on total system reward and complete ratio with online scheduling.

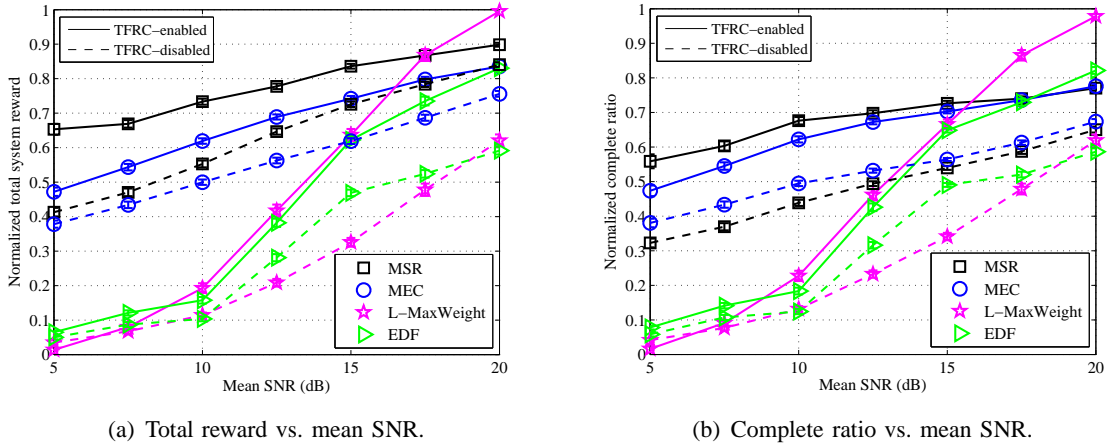


Fig. 8. Impact of mean SNR on total system reward and complete ratio with online scheduling.

lifetime. The performance loss in terms of the normalized total system reward is reduced by 30.4% for TFRC-disabled MSR when mean request lifetime increases from 10 s to 80 s, implying the benefits of extending request scheduling time. Besides, this improvement is more obvious if TFRC is enabled, e.g., the same performance loss can be further reduced by 60% for TFRC-enabled MSR when mean request life time changes within the same range. In Fig. 7, although MEC behaves different from other algorithms in terms of either normalized performance metric, the absolute values of both performance metrics with any of the compared algorithms increase as mean request lifetime extends. The dissimilarity is due to the different performance improvement speeds with respect to mean request time among those online and the offline DSFRB algorithms.

Fig. 8 shows the impact of channel quality on both total system reward and complete ratio with online scheduling. The normalized reward and complete ratio increase as the mean SNR

increases. This is because the improved channel quality greatly shortens the time to finish a request, which compensates for the negative impact of lacking accurate information on request demand and channel capacity, as compared with the offline DSFRB algorithm. Yet, as shown in the figure, when the mean SNR is larger than 17.5 dB, TFRC-enabled L-MaxWeight performs much better than the two newly proposed online algorithms. This is rational since the wireless network with the good channel quality can accommodate a traffic load larger than the one set in the simulations, i.e., corresponding to an underloaded system. Nevertheless, the proposed two algorithms perform much better when channel quality is poor or network traffic load is heavy.

REFERENCES

- [1] H. Shan, Y. Zhang, W. Zhuang, A. Huang, and Z. Zhang, "User behavior-aware scheduling based on time-frequency resource conversion," submitted to *IEEE Trans. Vehicular Technology*.
- [2] Y. Zhang, H. Shan, W. Zhuang, and A. Huang, "Time-frequency resource conversion based scheduling for on-demand data services," in *Proc. IEEE Globecom*, 2015.
- [3] H. Wu, X. Liu, and Y. Zhang, "Laxity-based opportunistic scheduling with flow-level dynamics and deadlines," in *Proc. IEEE WCNC*, 2013.